

REMARKS

Claims 1-29, 31-33, and 45-49 are pending and are rejected. Claims 1, 20, 23, and 45 are amended. Claims 17-18, 24, 31-33, and 47-48 are canceled. Reconsideration and allowance of Claims 1-16, 19-23, 25-29, 45-46, and 49 are respectfully requested.

Claim Rejections under 35 USC §103

Claims 1-4, 6, and 45-46 are rejected under 35 USC §103(a) as being unpatentable over “A Versatile Data String-Search VLSI,” written by Masaki Hirata and published by IEEE in April 1988, (hereinafter referred to as Hirata) in view of U.S. Patent No. 7,225,188 to Gai et al (Gai).

Claims 7-16, 19-21, and 31-33 are rejected as being unpatentable over Hirata in view of Gai and U.S. Patent 7,134,143 to Stellenberg et al (Stellenberg).

Claims 5 and 49 are rejected as being unpatentable over Hirata in view of Gai and U.S. Patent 6,963,942 to Igata (Igata).

Claims 17-18 are rejected as being unpatentable over Hirata in view of Gai and U.S. Patent 6,785,677 to Fritchman (Fritchman).

Claims 23-24 and 28-29 are rejected as being unpatentable over Hirata in view of Fritchman.

Claim 22 is rejected as being unpatentable over Hirata in view of Gai and Stellenberg, and in further view of “Fast Routing Table Look-up using CAMs,” written by Anthony J. McAuley and published by IEEE in 1993 (McAuley).

Claims 25-27 are rejected as being unpatentable over Hirata in view of Gai, Fritchman, and Stellenberg.

Claim 47 is rejected as being unpatentable over Hirata in view of Gai and McAuley.

Claim 48 is rejected as being unpatentable over Hirata in view of Gai and McAuley, and in further view of U.S. Patent 5,712,971 to Stanfill et al (Stanfill).

Applicant has amended the claims, and the patentability of each independent claim is discussed separately below.

Independent Claim 1

Applicant's Claim 1, as amended, recites:

A method, comprising:

receiving a text string having a plurality of characters; and

performing an unanchored search of a database of a stored patterns matching one or more characters of the text string using a state machine, wherein the state machine comprises a ternary content addressable memory (TCAM) and wherein the performing comprises comparing a state of the state machine and one of the plurality of characters with contents of a state field and a character field, respectively, stored in the TCAM, wherein the contents of the state field and the character field stored in the TCAM embody state transitions of the state machine, wherein the text string has zero or more wildcard characters, zero or more prefix characters preceding the wildcard characters and zero or more suffix characters succeeding the wildcard characters, and wherein performing the unanchored search comprises:

searching the database for a first pattern matching the prefix characters;

searching the database for a second pattern matching the suffix characters; and

creating a count that equals a number of the suffix characters plus a number of the wildcard characters.

The Office Action states that Hirata, Gai, and Fritchman disclose "creating a count that equals a number of the suffix characters plus a number of the wildcard characters (Hirata: pg: 330, col. 2 lns. 21-33 and Fritchman: col. 5, lns. 51-53 and 64-67)." Applicant disagrees.

The portion of Hirata cited by the Office Action states that "in the VLDC mode, data strings which start from a specified partial string and finish at another specified partial string can be searched," and "the length of the two specified partial data strings

and intermediate data string are variable.”¹ The language in Hirata fails to disclose “creating a count that equals a number of the suffix characters plus a number of the wildcard characters,” as recited in Claim 1.

The portion of Stellenberg referred to by the Office Action describes advancing an input data stream into an input register, and states “with this advancement, the oldest byte in the register 18 is shifted out, and a next byte in the data packet stream 14 is shifted in.”² Thus, in contrast to the Office Action’s assertion, Stellenberg does not disclose “creating a count that equals a number of the suffix characters plus a number of the wildcard characters, as recited in Claim 1.

Further, the Office Action has not pointed to any language in Gai that discloses creating a count that equals a *number of the suffix characters plus a number of the wildcard characters*.

Thus, Claim 1 is patentable over the cited references.

Claims 2-16, 19, and 22 depend from Claim 1 and therefore distinguish over the cited references for at least the same reasons as Claim 1.

Independent Claim 20

Applicant’s Claim 20, re-written in independent form, recites:

A method, comprising:

receiving a text string having a plurality of characters; and

performing an unanchored search of a database of a stored patterns matching one or more characters of the text string using a state machine, wherein the state machine comprises a ternary content addressable memory (TCAM) and wherein the performing comprises comparing a state of the state machine and one of the plurality of characters with contents of a state field and a character field, respectively, stored in the TCAM, wherein the contents of the state field and the character field stored in the TCAM embody state transitions of the state machine, wherein the performing further comprises:

1 Hirata: pg: 330, col. 2 lns. 25-33.

2 Stellenberg, col. 5, lines 61-67.

converging all branches of the state machine, for a given stored pattern, to a single next state when a first number of the characters are matched to the contents of a state field to all state transitions of the branches.

The Office Action states that Hirata, Gai, and Fritchman disclose “converging all branches of the state machine, for a given stored pattern, to a single next state when a first number of the characters are matched to the contents of a state field to all state transitions of the branches (Hirata: pg: 330, col. 2 lns. 21-33; pg. 331, col. 1, lns. 20-col. 2, lns. 4).” Applicant disagrees.

The first portion of Hirata cited by the Office Action describes how the length of the partial data strings can be variable, as discussed above with respect to Claim 1, and the second cited portion of Hirata discloses the circuit design for a 2-bit CAM cell and its operation. The language in Hirata thus fails to disclose “converging all branches of the state machine, for a given stored pattern, to a single next state when a first number of the characters are matched to the contents of a state field to all state transitions of the branches,” as recited in Claim 20.

Further, the Office Action has not pointed to any language in Gai or Stellenberg that discloses or suggests “converging all branches of the state machine, for a given stored pattern, to a single next state when a first number of the characters are matched to the contents of a state field to all state transitions of the branches,” as recited in Claim 20.

Thus, Claim 20 is patentable over the cited references.

Claim 21 depends from Claim 20 and therefore distinguishes over the cited references for at least the same reasons as Claim 20.

Independent Claim 23

Applicant's Claim 23, as amended, recites:

A method, comprising:

receiving a text string having a plurality of characters including a first number of prefix characters, a second number of wildcard characters succeeding the prefix

characters, and a third number of suffix characters succeeding the wildcard characters;

performing a first search on a ternary content addressable memory (TCAM) for a first stored pattern matching the prefix characters, wherein the first pattern stored in the TCAM includes state information indicative of a state machine and includes character information indicative of the first pattern; and

performing a second search of the TCAM for a second stored pattern matching the suffix characters, wherein the second pattern stored in the TCAM includes state information indicative of the state machine and includes character information indicative of the second pattern, wherein the performing further comprises:

creating a count that equals a number of the suffix characters plus a number of the wildcard characters.

Claim 23 is amended to recite that “the performing further comprises creating a count that equals a number of the suffix characters plus a number of the wildcard characters,” and is therefore patentable over the cited references for at least the same reasons as Claim 1.

Claims 25-29 depend from Claim 23 and therefore distinguish over the cited references for at least the same reasons as Claim 23.

Independent Claim 45

Applicant’s Claim 45, as amended, recites:

A string search apparatus, comprising:

control circuitry to receive a text string having a plurality of characters; and a pattern and state database including a ternary content addressable memory (TCAM) coupled to an associated memory, wherein the pattern and state database is operable to perform an unanchored search of the plurality of characters with patterns stored in the TCAM and associated memory by comparing a state of the state machine and one of the plurality of characters with contents of a state field and a character field, respectively, within the patterns stored in the TCAM, wherein the contents of the state

field and the character field stored in the TCAM embody state transitions of the state machine, wherein the control circuitry comprises:

- a first-in-first-out (FIFO) storage element;
- a register coupled to the FIFO storage element and the TCAM; and
- a rollback circuit coupled to the FIFO storage element.

The Office Action states that Stanfill discloses a “rollback circuit” at col. 10, lns 55-67. Applicant disagrees.

Stanfill is not concerned with string searching, but rather discloses a distributed database that can be easily managed. More specifically, Stanfill teaches a “method and system for running and checkpointing parallel and distributed applications which does not require modification to the programs used in the system nor changes to the underlying operating system.”³ The portion of Stanfill referred to by the Office Action describes using a “rollback command” to undo any changes made to the database.⁴ In contrast, Applicant’s Claim 45 discloses a “rollback circuit coupled to the FIFO storage element,” which in turn is part of “control circuitry to receive a text string having a plurality of characters.” Thus, because Stanfill fails to disclose a rollback circuit, the Office Action’s proposed combination of Stanfill with Hirata, Gai, and McAuley would not result in the structure recited in Applicant’s Claim 45. Accordingly, Claim 45 is patentable over the cited references.

Claims 46 and 49 depend from Claim 45 and therefore distinguish over the applied references for at least the same reasons as Claim 45.

³ Stanfill, col. 2, lines 13-17.

⁴ Stanfill states at col. 10, lines 48-51: The driver program 14 may issue a Rollback command, which will place the system in its state as of the start of the current phase, undoing any changes to files/databases, as described above.

CONCLUSION

In light of the above remarks, it is believed that Claims 1-16, 19-23, 25-29, 45-46, and 49 are allowable, and therefore a Notice of Allowance of Claims 1-16, 19-23, 25-29, 45-46, and 49 is respectfully requested. If the Examiner's next action is other than allowance as requested, the Examiner is requested to call the undersigned at (408) 236-6646.

Respectfully submitted,

A handwritten signature in black ink, consisting of a stylized 'W' followed by a long horizontal line extending to the right.

William L Paradise III
Patent Attorney
Reg. No. 38,990

April 21, 2008